```
entry:
 call void @llvm.dbg.value(metadata ptr %func_args, metadata !84, metadata
... !DIExpression()), !dbg !92
 %call = tail call i32 @initialise_arrays(ptr noundef nonnull @__func__.s253)
... #10, !dbg !93
 %call1 = tail call i32 @gettimeofday(ptr noundef %func_args, ptr noundef
... null) #10, !dbg !94
 tail call void asm sideeffect ".inst 0x2520e020", ""() #10, !dbg !95,
... !srcloc !97
 call void @llvm.dbg.value(metadata i32 0, metadata !86, metadata
... !DIExpression()), !dbg !98
 br label %for.cond2.preheader, !dbg !99
```

```
for.cond2.preheader:
 %nl.038 = phi i32 [ 0, %entry ], [ %inc21, %for.cond.cleanup4 ]
 call void @llvm.dbg.value(metadata i32 %nl.038, metadata !86, metadata
... !DIExpression()), !dbg !98
 call void @llvm.dbg.value(metadata i32 0, metadata !88, metadata
... !DIExpression()), !dbg !100
 %0 = call i64 @llvm.vscale.i64(), !dbg !101
 %1 = mul i64 %0, 2, !dbg !101
 %2 = icmp uge i64 8192, %1, !dbg !101
 br i1 %2, label %Pre.Vectorization, label
... %Preheader.for.remaining.iterations, !dbg !101
```

```
Pre.Vectorization:
 %7 = call i64 @llvm.vscale.i64()
 %8 = mul i64 %7, 2
 %step.vec = call <vscale x 2 x i64> @llvm.experimental.stepvector.nxv2i64()
 %9 = urem i64 8192, %8
 %total.iterations.to.be.vectorized = sub i64 8192, %9
 %10 = insertelement <vscale x 2 x i64> poison, i64 %8, i64 0
 %stepVector.update.values = shufflevector <vscale x 2 x i64> %10, <vscale x
... 2 x i64> poison, <vscale x 2 x i32> zeroinitializer
 br label %vectorizing.block
```

```
vectorizing.block:
 %11 = phi i64 [ 0, %Pre.Vectorization ], [ %25, %vectorizing.block ]
 %12 = phi <vscale x 2 x i64> [ %step.vec, %Pre.Vectorization ], [ %26,
... %vectorizing.block ]
 %13 = getelementptr inbounds [8192 x i32], ptr @a, i64 0, i64 %11, !dbg !116
 %14 = getelementptr inbounds [8192 x i32], ptr @b, i64 0, i64 %11, !dbg !124
 %15 = load <vscale x 2 x i32>, ptr %13, align 8
 %16 = load <vscale x 2 x i32>, ptr %14, align 8
 %17 = icmp sgt <vscale x 2 x i32> %15, %16
 %18 = getelementptr inbounds [8192 x i32], ptr @d, i64 0, i64 %11, !dbg !127
 %19 = getelementptr inbounds [8192 x i32], ptr @c, i64 0, i64 %11, !dbg !131
 %20 = call <vscale x 2 x i32> @llvm.masked.load.nxv2i32.p0(ptr %18, i32 8,
... <vscale x 2 x i1> %17, <vscale x 2 x i32> undef)
 %21 = mul <vscale x 2 x i32> %20, %16
 %22 = sub <vscale x 2 x i32> %15, %21
 %23 = call <vscale x 2 x i32> @llvm.masked.load.nxv2i32.p0(ptr %19, i32 8,
... <vscale x 2 x i1> %17, <vscale x 2 x i32> undef)
 %24 = add <vscale x 2 x i32> %22, %23
 call void @llvm.masked.store.nxv2i32.p0(<vscale x 2 x i32> %24, ptr %19, i32
... 8, <vscale x 2 x i1> %17)
 call void @llvm.masked.store.nxv2i32.p0(<vscale x 2 x i32> %22, ptr %13, i32
... 8, <vscale x 2 x i1> %17)
 %25 = add i64 %8, %11
 %26 = add <vscale x 2 x i64> %12, %stepVector.update.values
 %terminate.condition = icmp uge i64 %25, %total.iterations.to.be.vectorized
 br i1 %terminate.condition, label %middle.block, label %vectorizing.block
```

```
middle.block:
 %condition = icmp eq i64 %9, 0
 br i1 %condition, label %for.cond.cleanup4, label
... %Preheader.for.remaining.iterations
```

```
Preheader.for.remaining.iterations:
 %27 = phi i64 [ 0, %for.cond2.preheader ], [ %25, %middle.block ]
 br label %for.body5
```

```
for.body5:
 %indvars.iv = phi i64 [ %indvars.iv.next, %for.inc ], [ %27,
... %Preheader.for.remaining.iterations ]
 call void @llvm.dbg.value(metadata i64 %indvars.iv, metadata !88, metadata
... !DIExpression()), !dbg !100
 %arrayidx = getelementptr inbounds [8192 x i32], ptr @a, i64 0, i64
... %indvars.iv, !dbg !116
 %3 = load i32, ptr %arrayidx, align 4, !dbg !116, !tbaa !120
 %arrayidx7 = getelementptr inbounds [8192 x i32], ptr @b, i64 0, i64
... %indvars.iv, !dbg !124
 %4 = load i32, ptr %arrayidx7, align 4, !dbg !124, !tbaa !120
 %cmp8 = icmp sgt i32 %3, %4, !dbg !125
 br i1 %cmp8, label %if.then, label %for.inc, !dbg !126
```

```
if.then:
 %arrayidx14 = getelementptr inbounds [8192 x i32], ptr @d, i64 0, i64
... %indvars.iv, !dbg !127
 %5 = load i32, ptr %arrayidx14, align 4, !dbg !127, !tbaa !120
 %mul = mul nsw i32 %5, %4, !dbg !129
 %sub = sub nsw i32 %3, %mul, !dbg !130
 call void @llvm.dbg.value(metadata i32 %sub, metadata !85, metadata
... !DIExpression()), !dbg !92
 %arrayidx16 = getelementptr inbounds [8192 x i32], ptr @c, i64 0, i64
... %indvars.iv, !dbg !131
 %6 = load i32, ptr %arrayidx16, align 4, !dbg !132, !tbaa !120
 %add = add nsw i32 %sub, %6, !dbg !132
 store i32 %add, ptr %arrayidx16, align 4, !dbg !132, !tbaa !120
 store i32 %sub, ptr %arrayidx, align 4, !dbg !133, !tbaa !120
 br label %for.inc, !dbg !134
```

```
for.inc:
 %indvars.iv.next = add nuw nsw i64 %indvars.iv, 1, !dbg !135
 call void @llvm.dbg.value(metadata i64 %indvars.iv.next, metadata !88,
... metadata !DIExpression()), !dbg !100
 %exitcond.not = icmp eq i64 %indvars.iv.next, 8192, !dbg !136
 br i1 %exitcond.not, label %for.cond.cleanup4, label %for.body5, !dbg !101,
... !llvm.loop !137
```

```
for.cond.cleanup4:
 %call19 = tail call i32 @dummy(ptr noundef nonnull @a, ptr noundef nonnull
... @b, ptr noundef nonnull @c, ptr noundef nonnull @d, ptr noundef nonnull @e,
... ptr noundef nonnull @aa, ptr noundef nonnull @bb, ptr noundef nonnull @cc,
... i32 noundef 0) #10, !dbg !109
 %inc21 = add nuw nsw i32 %nl.038, 1, !dbg !110
 call void @llvm.dbg.value(metadata i32 %inc21, metadata !86, metadata
... !DIExpression()), !dbg !98
 %exitcond40.not = icmp eq i32 %inc21, 10, !dbg !111
 br i1 %exitcond40.not, label %for.cond.cleanup, label %for.cond2.preheader,
... !dbg !99, !llvm.loop !112
```

```
for.cond.cleanup:
 tail call void asm sideeffect ".inst 0x2520e040", ""() #10, !dbg !102,
... !srcloc !104
 %t2 = getelementptr inbounds %struct.args_t, ptr %func_args, i64 0, i32 1,
... !dbg !105
 %call23 = tail call i32 @gettimeofday(ptr noundef nonnull %t2, ptr noundef
... null) #10, !dbg !106
 %call24 = tail call i32 @calc_checksum(ptr noundef nonnull @__func__.s253)
... #10, !dbg !107
 ret i32 %call24, !dbg !108
```

CFG for 's253' function